

A Numerical Code for The Study of Water Droplets' Growth, Collision, Coalescence and Clustering
Inside Turbulent Warm Cloud-Clear Air Interfaces

Original

A Numerical Code for The Study of Water Droplets' Growth, Collision, Coalescence and Clustering Inside Turbulent Warm Cloud-Clear Air Interfaces / Ruggiero, V.; Codoni, David; Tordella, D.. - (2018). [10.5281/zenodo.2633679]

Availability:

This version is available at: 11583/2730442 since: 2019-04-09T16:36:56Z

Publisher:

Published

DOI:10.5281/zenodo.2633679

Terms of use:

openAccess

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)



A Numerical Code for The Study of Water Droplets' Growth, Collision, Coalescence and Clustering Inside Turbulent Warm Cloud-Clear Air Interfaces

V. Ruggiero^{a}, D. Codoni^b and D. Tordella^b*

^aCINECA, SCAI Rome

^bPolitecnico di Torino, DISAT

Abstract.

In past literature, most simulations of lukewarm clouds assumed static and homogeneous conditions. We are interested in simulating more realistic regimes of warm clouds that actually are systems which live in perpetual transitional situations. These time evolutions highly depend on the turbulent air flow hosting the cloud, and on transport phenomena taking place through the complex surfaces that bound the cloud with respect to the clear air surrounding it.

In our simulations, cloud boundaries (called interfaces in the text) are modelled through the shear-less turbulent mixing, matching two interacting flow regions - a small portion of cloud, and an adjacent clear air portion of equivalent volume - at different turbulent intensities. An initial condition reproduces local stable or unstable stratification in density and temperature. The droplets model includes evaporation, condensation, collision and coalescence. The typical water content inside a warm cloud parcel of about 500 m³, when associated to an initial condition, where drops are 30 microns in diameter, leads to an initial number of drops of the order of 10¹¹. A simulation grid up to 4092x2048x2048 points is sought after, which leads to a Taylor's microscale Reynolds number of 500. The governing equations are the Navier-Stokes equations under the Boussinesq's approximation, and are coupled to the transport equation for the water vapour represented as a passive scalar, and for drops seen as inertia particles, transported by background turbulence and gravity. The code uses a slab parallelization. The system contains a huge number of discrete elements, i.e. the water droplets, which undergo an intense clustering due to turbulent fluctuations. Turbulent clustering is not predictable, and in turn produces an imbalance on the communication rate among different cores. As a consequence, the computational burden among the cores in the cluster is not evenly distributed. This, per se, highly limits performance and binds the parallelization organization to that of a slab structure. Furthermore, clustering increases in time, and induces an inhomogeneous enhancement of the local droplets collision rate, as well as a concomitant depression of the growth in size of water droplets.

The long-term evolution of many kinds of transients must be considered in order to understand the above processes. This, in association to the variation of a quite large set of control parameters, will be the main motivation to ask in the future 0 level of computational resources for the simulation of water droplets' growth, collision, coalescence and clustering inside turbulent warm cloud-clear air interfaces.

1. Introduction.

Clouds are fugitive in nature. When looking at them for a few seconds, they seem to keep the same form. When looking again, after a minute, one finds that they are somewhat changed. Hardly, extended clouds formations then can live for more than 2-3 days. Their spatial structure is inhomogeneous with continuous changes associated with a large set of coexisting spatial and time scales. In previous literature, most simulations of lukewarm clouds in general assumed static and homogeneous conditions.

* Corresponding author. E-Mail address: v.ruggiero@ Cineca.it

Simulation of the microphysics and turbulence within clouds is an open problem. No single aspect has truly yet reached a level of standardization accepted by everyone in the community. In particular, the nucleation of the water droplet from Cloud Condensation Nuclei, CCN, that is the nucleation from aerosols of some kind: pollen, salt grains, soot, volcanic ash particles, smoke, dust of any kind, etc., has not yet found an efficacious representation inside Direct Numerical Simulation of Navier-Stokes-like turbulence. Notice that the aerosols' concentration (particles down to diameters of one nm) is too high (concentration of the order of 500 particles per cm^{-3}) for individual aerosols particles to be simulated as discrete elements. Actual computational resources do not allow a direct simulation of aerosols inside a cloud parcel. Thus, at the state of the art, it isn't possible to follow the entire process of droplet nucleation, condensation/evaporation, collision, coalescence (not to speak about possible fragmentation after collision) and related vapour transport and anisotropic clustering. This was evident from the communications and the discussion, developed at the recent Conference of the American Physical Society, Division of Fluid Dynamics, November 18-22, 2017, Denver, <http://www.apsdfd2017.org/> that hosted three Focus Sessions and a Mini-symposium on the Fluid Dynamics of Atmospheric Clouds (see references [1]) and also [2] Götzfried et al. JFM 2017).

An accurate simulation of the droplet growth, due to condensation in the oversaturated inner part of a cloud and due to collisions, which are enhanced by turbulence, is fundamental to understand the microphysics of clouds and therefore to explain how rain can be generated in warm clouds ([3] Grabowski-Wang, ARFM 2013, [4] Naumann-Seifert, J.Atmos.Sci.2016). Even if additional mechanisms, capable to further enhance droplet growth, have been proposed (e.g., condensation by radiative cooling at the cloud top, [5] Stevens et al., J. Atmos.Sci.,1996), turbulence plays a major role in these processes (see, e.g. [6] De Lozar-Muessle, Atmos.Chem.Phys. 2016). Therefore, accurate direct numerical simulations are central to understanding how rain can be initiated.

It should be noted that previous simulations on turbulence effects on collision efficiency have not yet provided data on sufficiently high Reynolds numbers (e.g., [7] Wang et al, NJP 2008, [8] Onishi et al.J.Atmos.Sci., 2015). Most studies are based on a small number of particles (10^5 - 10^8), which do not allow the investigation of rare extreme behaviours that might be important for rain formation. The insufficient knowledge of how rain is initiated in a stratocumulus rain, produces inadequate parameterizations of precipitation rates, with quite different results [9] (Wood, Mon.Weath.Rev., 2012).

High Reynolds number cloud parcel simulations, which include at least partially micro-physics droplets such as those here proposed, are at the frontier of current research (see e.g. the recent simulations by [8] Onishi et al., J.Atmos.Sci. 2015, the preliminary work by [10] Gotoh et al, NJP 2016 carried out at a relatively low resolution of 128^3 , and the simulations by [11] Ireland et al. JFM 2016). It should also be considered that the key observables we can calculate will also help to update sub-grid scale turbulence models to be used in large-eddy simulations of entire clouds (e.g. [12] De Lozar-Mellado, QJRM 2014).

In particular, the clustering (see figures 2 and 3) adds a further level of anisotropy to that already present inside the clear air-cloud interface, which is represented in our code by an unsteady shear-less turbulent mixing [13]. A high level of unsteady anisotropy means that different cores will execute each time step with a very uneven computational work, which is not predictable and thus it is not organisable *a priori*, from the computational point of view. In such a situation, where the physical modelling is under rapid evolution, it would be too early or premature to push the code to a massive level of parallelization. In fact, by shifting from slab to pencil parallelization (a parallelization structure already achieved for the version of the code where water droplets are not simulated), the time needed to exchange information between adjacent cores increases by a factor of 8. This is even more complex in a situation where the information exchanged by adjacent cores is certainly not homogeneous inside the computational domain, and where it is probable that nonadjacent cores must exchange information as well. This last eventuality is not *a priori* predictable, but has a high probability to happen, because turbulence hosts long-term phenomenon (bursts) which can induce large droplet displacements (i.e. droplet displacement to a domain portion seen by a core not adjacent to the core where the droplet departure took place) in time intervals so short that can be comparable with the single computational time step.

The two objectives sought during this PRACE Preparatory Access Type C grant were: i) the code optimization within the inherent physical complexity which leads to a highly discrete, uneven, unsteady distribution of water drops inside the domain portion seen by each core, ii) code scalability to a few thousands of cores. The activity carried out within this preparatory grant was useful to optimize some numerical features already present in the code, to learn how to profile the code source, to exploit the potentiality of advanced Fortran optimization and to reason - with the help of PRACE system administrators - on best strategies to obtain sufficiently good computational performances from the code, which is in continuous evolution. New ideas are in fact needed to extend the microphysical model of lukewarm clouds, which at the state of the art is far from being fully understood.

2. Code Description.

The simulation aims at reproducing the small-scale microphysical processes which occur inside a warm cloud and close to a cloud/clear air interface. The air motion is computed using an Eulerian formulation and the water droplet motion by a Lagrangian tracking method. In order to allow for evaporation/condensation processes of droplets, the humidity and temperature fields are calculated together with the momentum and mass balance equations. Therefore, the physical model includes:

- Navier-Stokes equations in the Boussinesq approximation: equations for mass, momentum, temperature (internal energy) of air, and the transport of water vapour seen as a passive scalar. This includes the feedback of water droplets on the flow (both momentum and energy) and the buoyancy (due to variations of temperature, humidity and liquid phase, e. g. [14] Kumar et al. J.Atmos.Sci. 2014).
- Lagrangian equations for the droplets, which are assumed to be spherical. Together with their momentum equation, an additional equation is solved for the droplet size variation as a function of the local relative humidity (e.g. [15] Vaillancourt et al., J.Atmos.Sci. 2001, 2002, [16] Pruppacher-Klett, Microphysics of Clouds and Precipitation, 1978).
- A module for the detection of collisions and the outcome of the collision ([17] Rabe et al., Phys.Fluids 2010).

Both versions of the code implementing the slab parallelization, the one without water droplet and that with water droplets, have been analyzed by means of a careful profiling with Intel VTune [18], Scalasca [19] and Intel Advisor [20].

For both versions, Intel VTune has been used to analyse the memory usage, and in particular, to identify memory-related issues. With Scalasca, we investigated the parallelization performance of the code to search for bottlenecks in the MPI parallelization, by increasing the size of the problem and the number of processes. Intel Advisor was used to identify loops that benefit from vectorization: in particular, to identify which aspect was blocking an efficient code vectorization and to find a relevant solution.

The optimization of the codes consisted of two steps; the first, related to the codes' general, and the second, to the porting of the codes on Intel KNL architecture.

For what concerns the first phase, the algorithm for the initial generation of the population of water droplet has been improved, and the initial conditions have now been obtained from the linear matching of the two initially homogeneous and isotropic fields inside a narrow region - as large as the flow integral scale - by means of a weighting function.

The NAG library has also been substituted with the FFTW library in order to improve performance. The last modification of the code concerns the I/O subroutines, as the old version of the code did not possess the ability to perform a restart.

For what concerns the second step, the porting of the codes for the Intel KNL architecture, to improve memory access performance and vectorization, both versions have been modified by reversing the order in a few loops. Vectorization has also been improved by adding compiler directives: forcing vectorization where this was inefficient and non-forcing it where the auto-vectorization was inefficient.

To evaluate the effects of the vectorization, we compared the performances between the vectorised code and the un-vectorised one, where the vectorization is disabled by compiling with the "no-vectorised" option. In the tables below, the wall clock time, measured in seconds, refer to the mean time step.

Code version, 2048 processes https://areeweb.polito.it/ricerca/philofluid/software.html	Not Vectorised	Vectorised
with droplets, v 1.4 p	130.2	127.5
without droplets, v1.4 & 1.7	15.0	5.7

Table 1 Wall clock time for average time steps (seconds).

It should be noted that a significant difference in performance between the two versions of the code is due to the routines `fftw.trasf1` and `aftw.trasf1`, that cannot be vectorised because of the way they access memory, which is specific to the physical structure of the simulated system. In the case without droplets, the computational weight of these routines is globally of the 2.5%. However, in the case with droplets the weight rises to 13.5 %.

Number of processes	Wall clock time	Normalized by the 64-processes Speed-up
64	153.5	1
128	78.0	0.98
256	45.7	0.89
512	23.1	0.83
1024	12.7	0.76
2048	5.7	0.87

Table 2 Data from the code without water droplets, grid 2048x2048x4096

Number of processes	Wall clock time	Normalized by the 64-processes Speed-up
64	1942.7	1
128	972	1
256	518.1	0.94
512	282.5	0.86
1024	166	0.73
2048	127.5	0.48

Table 3 Data from the code with water droplets, grid 2048x2048x4096

Strong scaling bottlenecks were identified by using Scalasca: in the pencil parallelized version without particles (see Table 2), the bottleneck was the barrier of MPI call, which is due to the load unbalancing among processes. In the slab parallelized version with particles (Table 3), the bottleneck was the increase of the communications among processes (in the MPI_Send_receive_replace call), which was associated both with the presence of the droplets and with the increase in the number of processes.

It should be noted that the code we use for water particles is not fit for weak scaling. The code simulates a 3D problem and it has a slab parallelization (along one of the homogenous spatial directions). The distribution direction is placed along one of the two statistically homogeneous directions (for the spatial air flow fluctuations). This feature could be exploited in the hope to have the maximum load balance among the processes. However, in order to have a fixed size of a problem per process by varying the number of processes, in this situation, only the size of the distributed direction can be changed. But in order to change the size of the problem, the size of all dimensions must be changed, not only that of the distributed one. Furthermore, it must be recalled that once water droplets are introduced into the system, the clustering phenomenology definitively spoils this homogeneity feature in the directions parallel to the mixing region, where cloud and clear air interact. As a consequence, the load among processes will be unbalanced unpredictably along the temporal evolution, as shown in Figure 2. It should be noted that the pencil parallelization would triple the communication time among processes, on average.

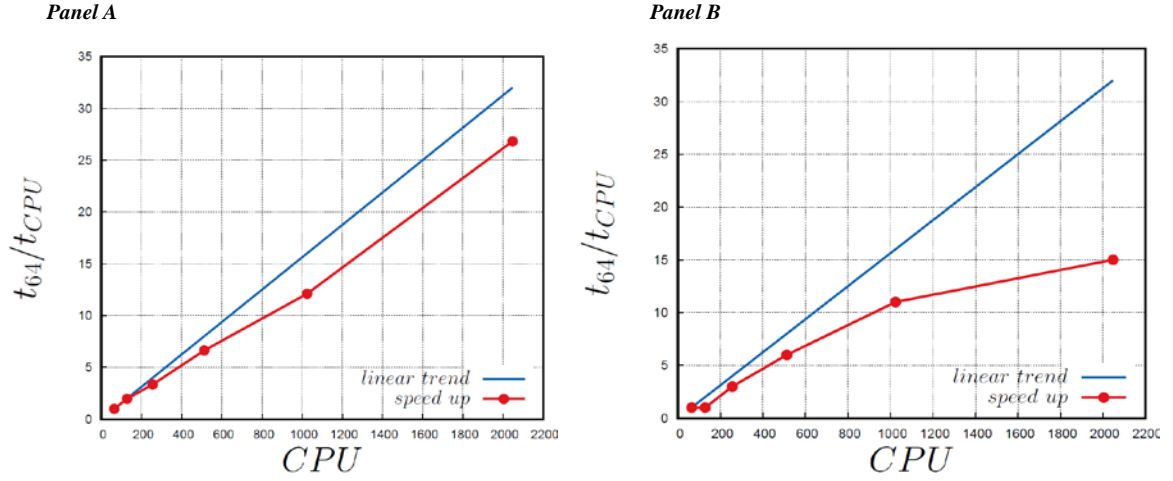


Figure 1 *Panel A*: speedup curve for the code version without water droplets. *Panel B*: speed up curve for the version with water droplets. (t_{64} execution time for 64 cores on CINECA Marconi KNL, t_{CPU} execution time for the cluster with the number of nodes specified in the ordinate, (computational grid 4096×2048^2).

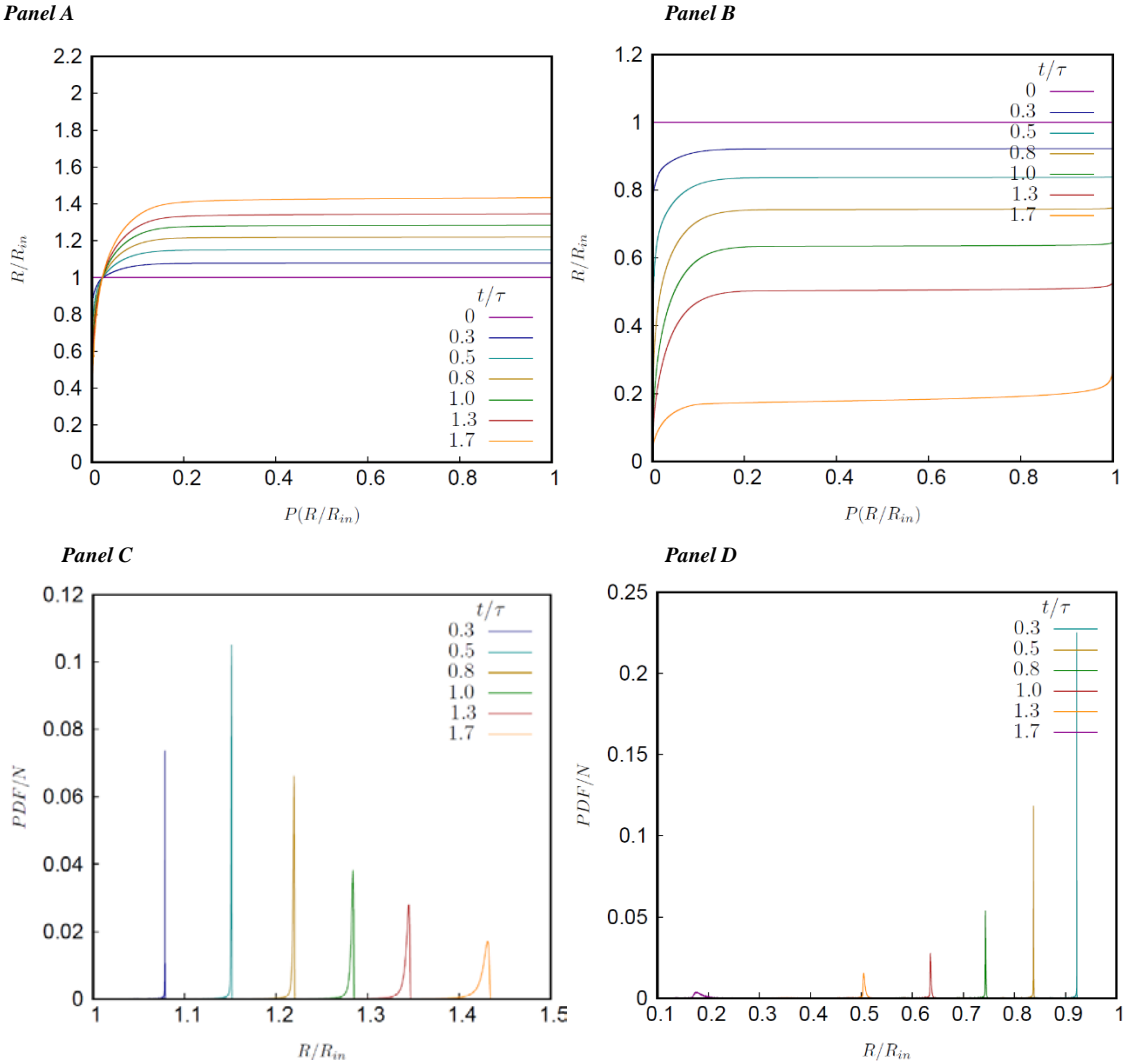


Figure 2. *Panels A and B*: cumulative density function of droplet diameters since the initial instant until 1.7 time scales in supersaturation and saturated cloud conditions, respectively. *Panels C and D*: evolution in time of the size distributions of the water droplet population. $Reynolds_\lambda$ 120, $512 \times 512 \times 1024$ grid points, integral scale 3 cm, cloud supersaturation 1.2, unstable stratification $Fr^2 = -30$. Four million of drops are simulated.

3. Some results and perspective computational needs

In the following, we discuss some examples of possible results that can be obtained by this kind of simulation. The results are relevant to a small grid of $512 \times 512 \times 1024$ points, because in this preparatory call we did not have resources to run the long-term simulations on the largest grid ($2048 \times 2048 \times 4096$), which we wish to use in the future, to study a parcel of cloud with a linear dimension of the order of 10 meters. To check the scalability of the codes, we actually ran just a few simulations with the largest grid size over a small number (order of 10^2) of integration time steps. However, in such cases, results are not yet physically relevant and cannot be shown here.

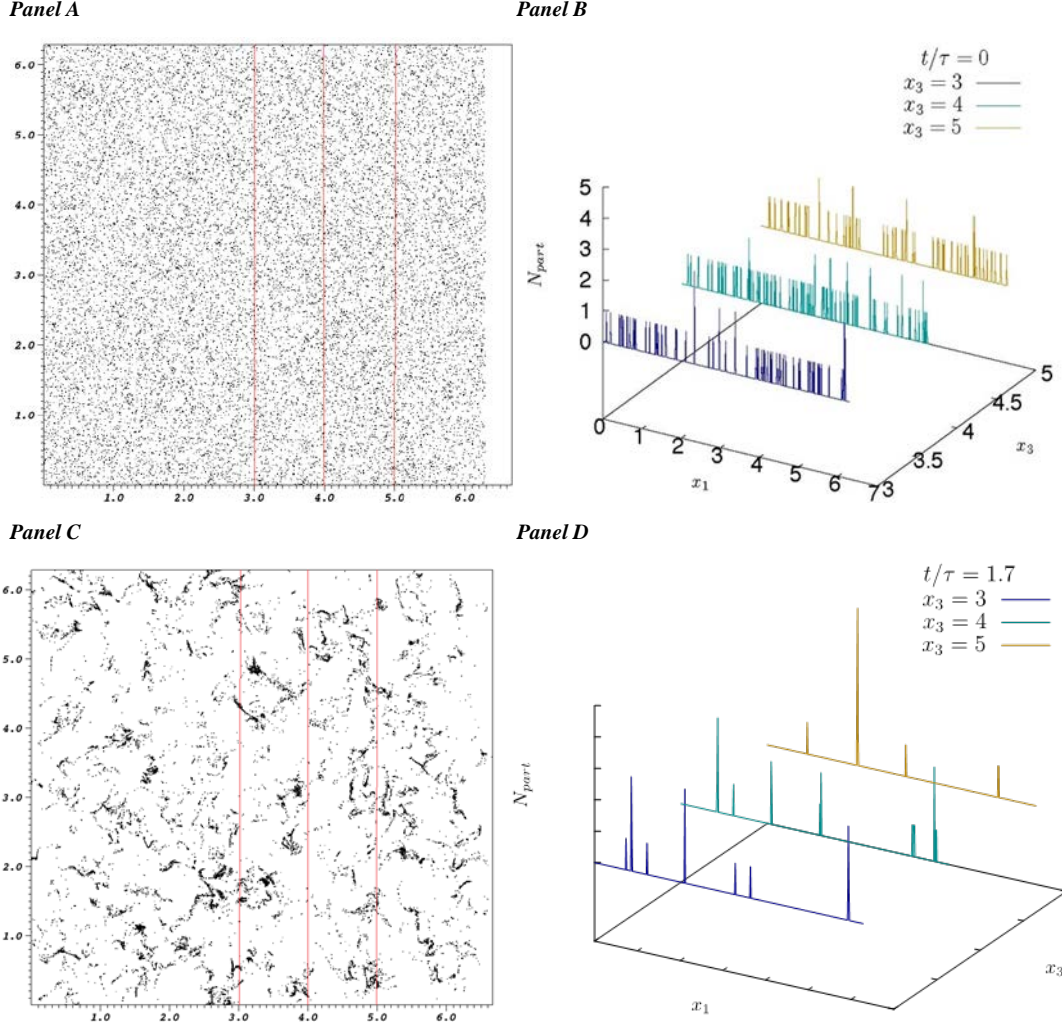


Figure 3, *Panel A*: initial distribution of drops inside the cloud region. Physical parameters as described in the caption of Figure 2. Only the cloud region is here visualized, for simplicity the clear air part is not included. *Panel B*: visualization of the droplet concentration spikes after clustering along one of the directions parallel to the interface across which clear air and cloud interact. Three sections along x_3 are shown, x_3 being the gravity direction which is supposed normal to the interface. The yellow line represents the concentration inside the interaction region. *Panel C*: distribution of drops inside the cloud region after 1.7 eddy turn over times. *Panel D*: same as above, after 1.7 time scales.

For both supersaturation and saturation conditions of the cloud region, Figure 2 shows information on the size distribution of the droplets after one eddy turn over time, within which they could grow/decay via condensation/evaporation, collision (also multiple collisions are considered), and coalescence. One can see that the growth in size is accompanied by an enlargement and increased skewness of the size distribution, a fact which was also observed in recent laboratory experiments. The laboratory simulation carried out at the Michigan State University, in fact shows that this trend must be accompanied by a decrease in the aerosol concentration, leading to the droplet nucleation ([21] Chandrakar et al. PNAS, 2016). It's worth noting that this part of the physics, nucleation dynamics, is not yet included in the code. To understand how to implement such a fundamental part of physics, we will exploit the results yield by the transients which we foresee to study in the next run campaign where we foresee to simulate a portion of cloud with a linear dimension of $1/10$ the integral scale of a typical warm cloud (about 100 m). For these we intend to ask PRACE Tier 0 resources.

Under the hypothesis that computational cells can be considered as a closed system in relation to the typical time scale of the nucleation activation, we foresee, at the sub-grid level, to implement the solution of the droplet population balance equation parametrised on the initial concentration of activation nuclei and relevant initial time

scale ([22] Codoni et al., EGU 2018).

Figure 3 highlights the intense droplet clustering phenomenon. As described above in section 2, clustering is the main cause of the unbalanced communications among the cores. Clustering increases along the temporal evolution of the system, and of course, is not predictable. Consider that here we simulated four million of droplets only, which is a small number with respect to the number actually contained in the liquid water content inside a lukewarm cloud of equivalent volume.

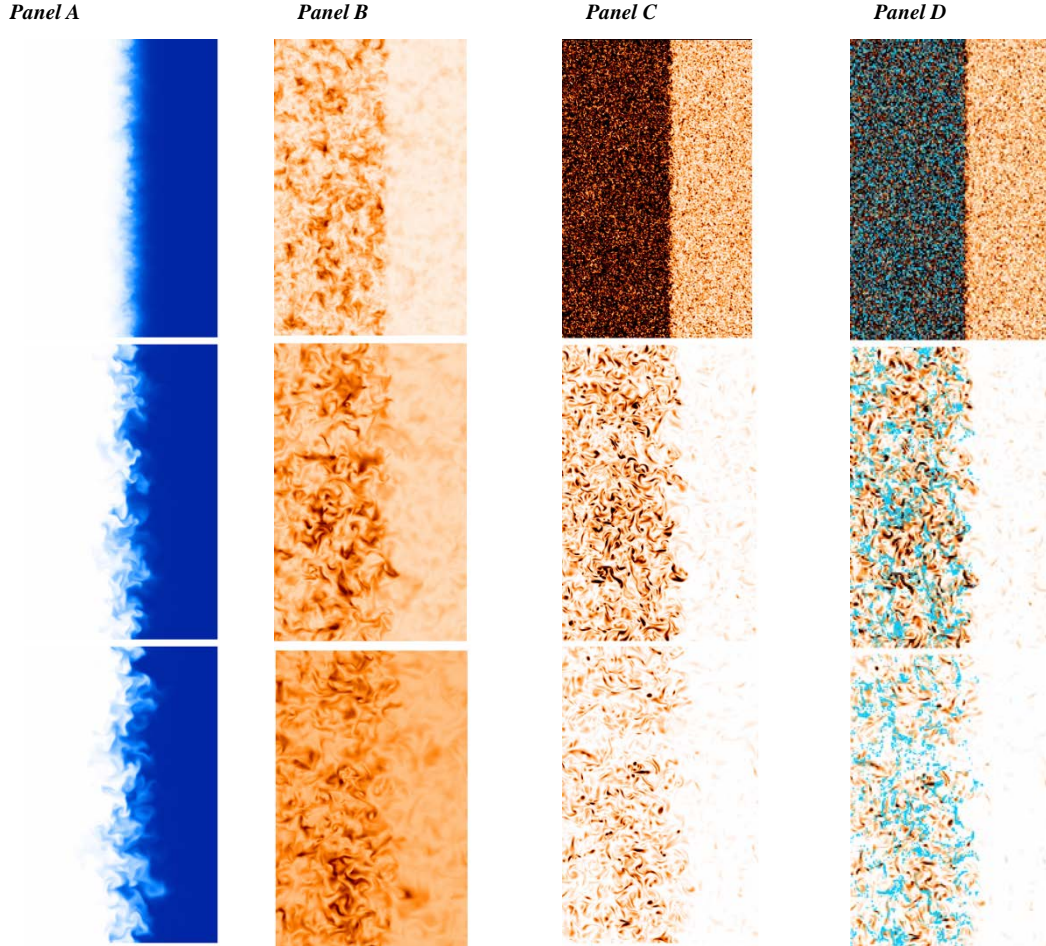


Figure 4 *Panel A*: visualization of the vapour density across the interface clear air and cloud. *Panel B*: kinetic energy dissipation of the air fluctuations across the boundary of the cloud. *Panel C*: visualization of the enstrophy of the air fluctuation across the cloud boundary. The images refer to three different instants from the beginning of the simulation and one eddy turn over time. Physical parameters as described in the caption of figure 2. *Panel D*: same as previous one, but with water droplets visualized in light blue.

Finally, Figure 4 characterizes the background air flow fluctuations of the cloud – clear air interface. It should be recalled that the droplets are actually evolving in the left parts of each field visualized in Figure 4. The large grid (2048X2048X4096) we aim at is the minimum size that can host a simulation representing the last three wavenumber decades associated to the turbulence cascade that precede the dissipation range of lukewarm clouds. This kind of clouds have an integral scale of the order of 100 meters and a Kolmogorov microscale of the order of a few mm. We aim at simulating the range of scales from 10 m to 1 mm, i.e. wave numbers in the range $6 \cdot 10^{-1} - 6 \cdot 10^3 \text{ m}^{-1}$ (four decades), while performing a set of initial value problem representing different possible situations met in clouds' lives: - stable and unstable density stratification, - level of super- or sub-vapour saturation, - ratio of space macroscales in the turbulence inside and outside the cloud, - ratio of turbulent kinetic energy inside and outside the cloud, Reynolds number, - aerosol concentration activating droplet nucleation. At this stage of knowledge, the important goal in our opinion is the exploration of these transients over time intervals of sufficient lengths to understand the links between the different facets of the involved phenomenology. The great amount of the needed computational resources is therefore related to the length of these transients, which must be at least of the order of 20-30 eddy turn over times.

References

- [1] Conference of the American Physical Society, Division of Fluid Dynamics, November 18-22, 2017, Denver, <http://www.apsdfd2017.org/>, Bulletin APS DFD, Vol.62, Number 14, Focus Sessions on the Fluid Dynamics of Atmospheric Clouds: I, G15, pages 322-324, II L16, pages 431-433, II M15, pages 505-507. See, among many other for example, the abstracts:
 - 3.1 G15 4 TC. Basso, M. Iovieno, S. Bertoldo, G. Perotto, A. Athanasia., G. Perona, D. Tordella, Green, disposable, mini radiosondes to track fluctuations along isopycnic surfaces in cloud environments.
 - 3.2 G15 6 Abade G, W. Grabowski, H. Pavloska, Broadening of cloud droplet spectra through turbulent entrainment and eddy hopping.
 - 3.3 G15 7 P. Goetzfried, B. Kumar, R. Shaw, DNS of cloud-clear air mixing at a shear-free interface
 - 3.4 A. Keirstein, S. Krueger, Enhancement of coalescence in turbulent clouds
 - 3.5 I. Saito, T. Gotoh, T. Watanabe, Growth of cloud droplets from aerosol in turbulence.
- [2] P. Götzfried, B. Kumar, R.A. Shaw, J. Schumacher (2017), *J. Fluid Mech.* 814, pp. 452-483.
- [3] W.G. Grabowski and L.P. Wang (2013), *Ann. Rev. Fluid Mech.* 45, pp. 293-324.
- [4] A.K. Naumann and A. Seifert (2016), *J. Atmos. Sci.*, 73, pp. 2279–2297.
- [5] B. Stevens, G. Feingold, W. Cotton, and R. Walko (1996), *J. Atmos. Sci.* 53, pp. 980-1006.
- [6] A. de Lozar and L. Muessle (2016), *Atm. Chem. Phys* 16, pp. 6563-6576.
- [7] D.W. Wang (2008), *NJP* 10.
- [8] Onishi, R., K. Matsuda, and K. Takahashi (2015), *J. Atmos. Sci.*, 72, pp. 2591–2607.
- [9] R. Wood (2012), *Mon. Wea. Rev.*, 140, pp. 2373–2423.
- [10] T. Gotoh, T. Suehiro and I. Saito (2016), *New Journal of Physics*, 18(4), 043042, pp. 1-19.
- [11] P. Ireland, A. Bragg and L. Collins (2016), *J. Fluid Mech.* 796, pp. 659-711.
- [12] A. de Lozar, and J.P. Mellado (2014), *Q.J.R. Meteorol. Soc.* 140, pp. 1493–1504.
- [13] (a) D. Tordella and M. Iovieno (2011), *Phys. Rev. Lett.* 107, 194501, (b) D. Tordella, M. Iovieno, and P.R. Bailey (2008), *Phys. Rev. E* 77, 016309. (c) L. Gallana, S. Di Savino, F. De Santi, M. Iovieno, D. Tordella (2014), In: 32nd UIT Heat Transfer Conference, Pisa, 23-25 June 2014.
- [14] B. Kumar, J. Schumacher and R.A. Shaw (2014), *J. Atmos. Sci.* 71, pp. 2564–2580.
- [15] P.A. Vaillancourt, M.K. Yau and W. W. Grabowski (2001, 2002), *J. Atmos. Sci.*, 58, 59, pp. 1945–1965, pp. 3421–35.
- [16] H.R. Pruppacher and J.D. Klett (1978), “*Microphysics of Clouds and Precipitation*”, Springer Netherlands.
- [17] C. Rabe, J. Malet, and F. Feuillebois (2010), *Physics of Fluids* 22, 047101.
- [18] Intel VTune, “<https://software.intel.com/en-us/vtune-amplifier-cookbook>”
- [19] Scalasca, “<http://www.scalasca.org/>”
- [20] Intel Advisor, “<https://software.intel.com/en-us/advisor>”
- [21] K.K. Chandrakar, W. Cantrell, K. Chang, D. Ciochetto, D. Niedermeier, M. Ovchinnikov, R.A. Shaw, F. Yang (2016), *PNAS*, vol. 113, No. 50, pp. 14243-14248.
- [22] D. Codoni, M. Golshan, V. Ruggiero, M. Vanni and D. Tordella (2018), EGU.

Acknowledgements

This work was financially supported by the PRACE project funded in part by the EU’s Horizon 2020 Research and Innovation programme (2014-2020) under Grant Agreement 730913 and by the network H2020 MSCA ITN ETN COMPLETE 675675.